



**University of
Zurich**^{UZH}

**Zurich Open Repository and
Archive**

University of Zurich
University Library
Strickhofstrasse 39
CH-8057 Zurich
www.zora.uzh.ch

Year: 2010

Pixel detector data quality monitoring in CMS

Rose, K ; Blekman, F ; Chiochia, V ; Gomez-Ceballos, S H C ; Merkel, P

Abstract: The silicon pixel detector in the Compact Muon Solenoid (CMS) at the Large Hadron Collider contains approximately 66 million channels, and will provide extremely high tracking resolution for the experiment. To ensure the data collected is valid, it must be monitored continuously at all levels of acquisition and reconstruction. The Pixel Data Quality Monitoring (DQM) process ensures that the detector, as well as the data acquisition and reconstruction chain, is functioning properly. The monitoring process is designed such that it can examine the pixel detector with high enough granularity that potential problems can be identified and isolated, while running efficiently enough that action can be taken before much data is compromised.

DOI: <https://doi.org/10.1088/1742-6596/219/3/032056>

Posted at the Zurich Open Repository and Archive, University of Zurich

ZORA URL: <https://doi.org/10.5167/uzh-45283>

Journal Article

Published Version



The following work is licensed under a Creative Commons: Attribution 4.0 International (CC BY 4.0) License.

Originally published at:

Rose, K; Blekman, F; Chiochia, V; Gomez-Ceballos, S H C; Merkel, P (2010). Pixel detector data quality monitoring in CMS. *Journal of Physics: Conference Series*, 219(3):032056.

DOI: <https://doi.org/10.1088/1742-6596/219/3/032056>

Pixel detector data quality monitoring in CMS

This article has been downloaded from IOPscience. Please scroll down to see the full text article.

2010 J. Phys.: Conf. Ser. 219 032056

(<http://iopscience.iop.org/1742-6596/219/3/032056>)

View [the table of contents for this issue](#), or go to the [journal homepage](#) for more

Download details:

IP Address: 89.206.70.240

The article was downloaded on 13/02/2011 at 16:00

Please note that [terms and conditions apply](#).

Pixel Detector Data Quality Monitoring in CMS

Keith Rose¹, Freya Blekman², Vincenzo Chiochia³, Shan-Huei Chuang Gomez-Ceballos¹, Petra Merkel⁴

¹Rutgers University, Piscataway, New Jersey, USA, ²Cornell University, Ithaca, U.S.A.,
³University of Zurich, Zurich, CH, ⁴Purdue University, West Lafayette, USA

E-mail: keith.jarid.rose@cern.ch

Abstract. The silicon pixel detector in the Compact Muon Solenoid (CMS) at the Large Hadron Collider contains approximately 66 million channels, and will provide extremely high tracking resolution for the experiment. To ensure the data collected is valid, it must be monitored continuously at all levels of acquisition and reconstruction. The Pixel Data Quality Monitoring (DQM) process ensures that the detector, as well as the data acquisition and reconstruction chain, is functioning properly. The monitoring process is designed such that it can examine the pixel detector with high enough granularity that potential problems can be identified and isolated, while running efficiently enough that action can be taken before much data is compromised.

1. Introduction

The Pixel Data Quality Monitoring process is run as one of 15 subprocesses of the CMS central DQM [1]. Events are processed (including reconstruction) at a rate of 5 to 10 Hz, as part of a dedicated DQM stream of monitored data. The goal of the process is to produce and store a series of histograms using a framework known as *root* [2]. The histograms are selected such that they properly encapsulate the status of the pixel detector over the course of a run. The variables tracked, as well as the thresholds that define good data are set in a series of XML files read by the application at startup. A brief list of most relevant variables can be found in Table 1. The application itself is written in C++ within the CMS software framework (CMSSW) [3], and runs in trivial time compared to preceding reconstruction.

Table 1. List of variables tracked by the Pixel DQM

Reconstruction Level Name	variables
Raw Data	Errors from data unpacking and readout
Digis (digitized hits)	raw charge and pixel position
Clusters	calibrated charge, size and position
RecHits	hit position after Lorentz angle correction
Tracking	track residuals in x and y, on/off track clusters

The standard output of the DQM process is a *root* file on the order of 20 MB, including some information on all 66 million pixels and all 1440 modules of the detector [4] (a module contains between 8000 and 66000 pixels). The output files are archived for future viewing as a part of a standard procedure common to all CMS subdetectors. The results may also be written to an online database.

2. Online Tools

The first level of feedback to the central DQM and offline Pixel shift crew are the Pixel Summary Map shown above in Figure 1 and shift report histograms gathered in the central online GUI for easy browsing. These are produced via processes running off the dedicated DQM data stream. The online browser [5] makes a series of XMLHTTP requests for various monitor elements, and provides the necessary plots on the order of every 1 to 3 minutes. During online processing, shifters have access to the full pixel granularity, including hitmaps for individual modules.

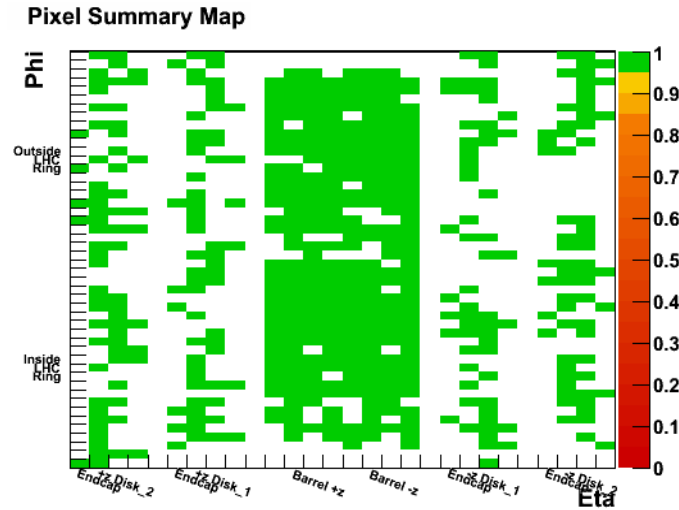


Figure 1. Pixel Summary Map, the first level of online DQM feedback to shifters. Each bin represents up to three modules, with the value scaling from 0 to 1 in increasing data quality.

Over time, as we continue to understand optimal data-taking conditions for the pixel detector, we will be able to identify and set alarm thresholds for all of the variables covered by the DQM process. These will then be assigned appropriately to trigger alarms in the shift histograms, and the effect on the pixel detector state will be represented in the Pixel Summary Map. Optimally, the map is designed to provide maximum feedback in a single plot. Each bin of the histogram represents up to three modules, with the filled value varying between 0 and 1 depending on the overall adherence of the data to norms.

A pixel specific java-based web GUI exists, using AJAX for web services, allowing shifters to browse histograms read from the dedicated DQM data stream, indexed by pixel detector substructure, as shown in Figure 2. Alarm thresholds are set, again based on XML files read by the client process, and applied to the displayed histograms so that problems identified in the shift histograms and summary map can easily be traced to a specific pixel detector subsection.

Additional applications (optional for shifters, some still under development) include an SVG map of the entire tracker, allowing point-and-click browsing of the individual modules, as shown in Figure 3.

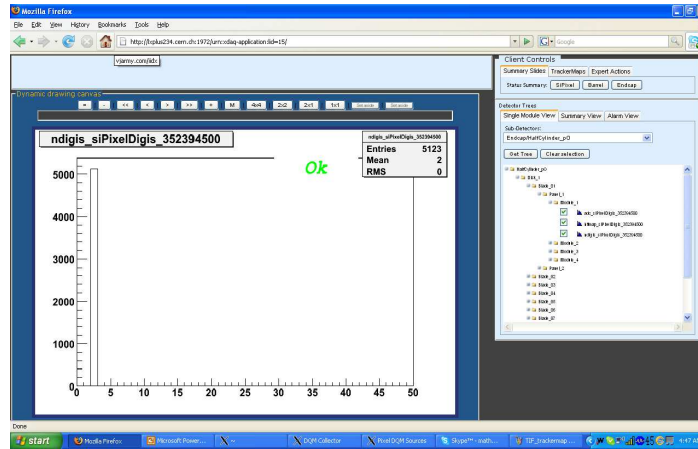


Figure 2. Pixel DQM GUI, allowing users to inspect all histograms in the data stream for a given run. Available for both online and offline processing.



Figure 3. SVG pixel detector map. A visual representation of the pixel detector, this single-page overview allows problems to be traced to a physical location in the pixel detector. Relevant histograms can be retrieved with a simple point-and-click interface.

3. Offline Tools

The second level of feedback comes from the offline Pixel DQM processing, which runs in parallel with the central DQM. Workers on the offline Pixel shift study fully calibrated and reprocessed data with full statistics. Again, the java-based GUI is used for this purpose. To cut down on the number of histograms that are stored in memory during the central DQM process, the Pixel DQM cuts off granularity one level above the maximum, reducing the file size by about a factor of 10.

Once significant data has been taken and the performance of the pixel detector is well understood, the offline shifter will have reference plots available to compare to the output of the DQM process to confirm the quality of a given run. As well, scripts will be available to apply quality test thresholds similar to those applied in the GUI to allow rapid processing of data.

4. Historical Trend Monitoring

Beyond the need to monitor the pixel detector on a run-by-run basis, the Pixel DQM shift crew must be able to check the behavior of detector subsections over a period of time. For this purpose, we have implemented historical trend monitoring as part of the offline shift duties. The historical monitoring program runs on the *root* output of the Data Quality Monitoring process. The data is harvested on a run-by-run basis and stored in a pixel database object. The object stores on the order of 100 floating point numbers per module, and takes up disk space on the order of 1 MB per run stored.

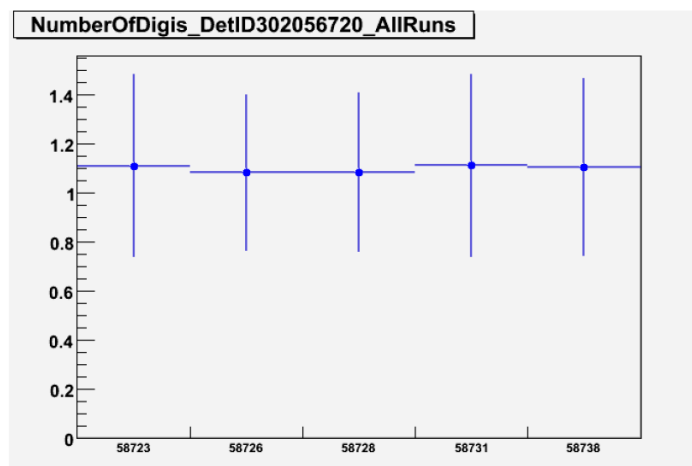


Figure 4. Simple preliminary plot produced by historic trend monitoring application, in this case number of Digis over a series of runs.

Histogram representations of various historical trends are also output via *root*, such as one shown in Figure 4. Various tools are available to extract the information from the historic database, plot the trends, and eventually host them for viewing on the web.

5. Pixel Detector Commissioning

The various tools used to calibrate the Pixel detector have been designed to produce output analogous to that of the standard Data Quality Monitoring. Thus, calibration can be run alongside the DQM process using the same series of XML requests. The output of these can later be stored in a database object containing calibration constants.

As part of the standard series of monitoring elements, the Pixel DQM also records the number of events each module appears in, as well as an occupancy map for all modules in one portion of the Pixel detector overlaid on each other. Areas which are unusually noisy can be isolated using information from these two monitoring elements, allowing noisy pixels to be observed in real time. As of March 2009, approximately 260 static noisy pixels have been found and masked.

6. Conclusion

The CMS Pixel detector is a highly complex device which will allow physicists to investigate LHC collisions in great detail. Monitoring data quality from 66 million channels is a formidable task. We have created a suite of monitoring programs and procedures that will help us ensure the Pixel detector lives up to its promise.

7. References

- [1] Meyer, A 24 Feb 2009 *DQM Subdetectors* <https://twiki.cern.ch/twiki/bin/view/CMS/DQMSubDetectors>
- [2] ROOT Development Team *ROOT: A Data Analysis Framework* root.cern.ch
- [3] CMS Collaboration *Detector and Physics Performance TDR CERN/LHC 2006-001* p 28
- [4] CMS Collaboration *Detector and Physics Performance TDR CERN/LHC 2006-001* p 19
- [5] Tuura L 19 Mar 2009 *DQM Gui Gallery* <https://twiki.cern.ch/twiki/bin/view/CMS/DQMGuiGallery>